Homework #1

Due Date: 11:59pm Saturday 22 March 2014

Write a C++ program that can read and write PPM image files and perform some simple color manipulations on them. Your program will read the image on `stdin` and outputs the resulting image on `stdout`, and get in the required operation on the command line.

1. [6 points] Modify the file `PPM.cpp` and add the implementation for:
    1. [1 point] Data members to store the information for a PPM file.
    2. [1 point] A function to read in the PPM file format described below from `stdin` and put the result in a member 2D array of `unsigned char`'s.
    3. [1 point] A function to write the contents of a structure containing a PPM file onto `stdout`.
    4. [1 point] A function to substitute one color with another color in the whole picture.
    5. [2 points] A function to convert color RGB pixels (files with header P3) into grayscale images (files with header P2). The grayscale value for each pixel is defined as:
       `Gray = 0.298839*R + 0.586811*G + 0.114350*B`
2. [2 points] Modify the file `main.cpp` to parse in the inputs and options, perform the operation, and write the output.
3. [1 point] How many bytes should the size of `checker.ppm` be? How many bytes is it actually? Why is the actual size different from the supposed size?

PPM File

The PPM file is a text file representation for images. It has several formats depending on whether the image is binary, grayscale, or color. For more information, check https://en.wikipedia.org/wiki/Netpbm_format.

The color PPM format is defined as:

```
P3
[xres] [yres]
[max intensity]
[r0] [g0] [b0]
[r1] [g1] [b1]
….
```

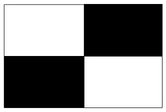
where (r0, g0, b0) is the color of the top-left pixel, and pixels go from the top-left towards the bottom right.

For example:

```
P3
2 2
255

255 255 255
0 0 0
0 0 0
255 255 255
```

represents a 2x2 image (with 4 pixels) where the top-left pixel is white, the top-right is black, the bottom-left is black, and the bottom-right is white.



The grayscale PPM file is defined as:

```
P2
[xres] [yres]
[max intensity]
[g0] [g1] [g2] ...
```

where g0 is the grayscale value of the top-left pixel, and pixels go from the top-left towards the bottom right. So, the same image above would be written as:

```
P2
2 2
255
255 0 0 255
```

To view PPM files, you can use the display utility (part of ImageMagick) as follows:

```
      display image.ppm
```

Command Line

Your program should be named hw0, and should be called as follows:

- To substitute colors:

  ```
  ./hw01 -sub r0 g0 b0 r1 g1 b1 < input.ppm > output.ppm
  ```

  where `(r0, g0, b0)` is the input color that should be replaced by `(r1, g1, b1)`, the input image is called `input.ppm` and the output is written into the image `output.ppm`. For example, to convert the green color in `checker.ppm` into black, you could run:

  ```
  ./hw01 -sub 0 255 0 0 0 0 < checker.ppm > output.ppm
  ```

- To convert RGB images to grayscale images:

  ```
  ./hw01 -rgb2gray < input.ppm > output.ppm
  ```

  where the input image is called `input.ppm` and the output is written into the image `output.ppm`. For example, to convert the image `checker.ppm` into grayscale, you could run:

  ```
  ./hw01 -rgb2gray < checker.ppm > output.ppm
  ```

Instructions

- All code should be implemented in C++ under Linux.
- Please submit your homework in one zip file named as follows: *CMPN206.HW##.FirstName.LastName.zip*, so for example if your name is Mohamed Aly and this is homework #1, then the file name should be *CMPN206.HW01.Mohamed.Aly.zip*.
- Please include all your code and sample output in the zip file, with a README file to explain what you did. Failure to follow these instructions will cause deductions from your grade.
- You are allowed to discuss the problems among yourselves. However, **copying** any part of the code will result a grade of **ZERO**. No exceptions.

Grading

- 9 points: requirements above
- 1 point: submission instructions